Pipelined A/D Converters

- Ideal operation
- Errors and correction
  - Redundancy
  - Digital calibration
- Implementation
  - Practical circuits
  - Stage scaling
Block Diagram

- Idea: Cascade several low resolution stages to obtain high overall resolution
- Each stage performs coarse A/D conversion and computes its quantization error, or "residue"

Characteristics

- Number of components (stages) grows linearly with resolution
- Pipelining
  - Trading latency for conversion speed
  - Latency may be an issue in e.g. control systems
  - Throughput limited by speed of one stage → Fast
- Versatile: 8...16bits, 1...200MS/s
- Many analog circuit non-idealities can be corrected digitally
Concurrent Stage Operation

- Stages operate on the input signal like a shift register
- New output data every clock cycle, but each stage introduces ½ clock cycle latency

Data Alignment

- Digital shift register aligns sub-conversion results in time
Pipelined ADC Analysis

• Ignore timing and use simple static model

\[ V_{in} \xrightarrow{B_1 \text{ Bits}} V_{res1} \xrightarrow{B_2 \text{ Bits}} \ldots \xrightarrow{B_k \text{ Bits}} D_{out} \]

• Let's first look at "two-stage pipeline"
  – E.g.: Two cascaded 2-bit ADCs to get 4 bits of total resolution
Two Stage Example

- Using only one ADC: output contains large quantization error
- "Missing voltage" or "residue" ($-\epsilon_{q1}$)
- Idea: Use second ADC to quantize and add $-\epsilon_{q1}$

Two Stage Example

- Use DAC to compute missing voltage
- Add quantized representation of missing voltage
- Why does this help? How about $\epsilon_{q2}$?
Two Stage Example

- Fine ADC is re-used $2^2$ times
- Fine ADC’s full scale range needs to span only 1 LSB of coarse quantizer

$$
\epsilon_q = \frac{V_{ref2}}{2^2} = \frac{V_{ref1}}{2^2 \cdot 2^2}
$$

Two Stage Pipelined ADC Transfer Function

- Coarse Bits (MSB)
- Fine Bits (LSB)
Two Stage (2+2) Pipelined ADC

Cascading More Stages

- LSB of last stage becomes very small
- Impractical to generate several $V_{\text{ref}}$
Gain Elements

- Practical pipelines use single $V_{\text{ref}}$
- Precision requirements decrease down the pipe
  - Advantageous for noise, matching (later)

Complete Pipeline Stage

"Residue Plot"
E.g.:
$B = 2$
$G = 2^2 = 4$
Errors

- We cannot build perfect ADCs, DACs and gain elements
- How can we tolerate/correct errors?
- Let's first look at sub-ADC errors
- Assumptions:
  - Ideal DAC, ideal gain elements

ADC Model

\[
D_{out} = V_{in,ADC} + \varepsilon_d \left( 1 - \frac{G_1}{G_{d1}} \right) + \varepsilon_{g2} \left( 1 - \frac{G_2}{G_{d2}} \right) + \ldots \left( 1 - \frac{G_{(n-1)}}{G_{d(n-1)}} \right) + \varepsilon_{qn} \left( \frac{1}{\prod_{j=1}^{n-1} G_d} \right)
\]
ADC Model

- If the "Analog" and "Digital" gains match exactly, we get:

\[
D_{out} = V_{in,ADC} + \frac{\varepsilon_{qn}}{\prod_{j=1}^{n-1} G_j}
\]

\[
B_{ADC} = B_n + \sum_{j=1}^{n-1} \log_2 G_j
\]

Observations

- The aggregate **ADC resolution is independent of sub-ADC resolution**
- **Effective** stage resolution \(B_j = \log_2(G_j)\)
- **Conversion error does not (directly) depend on sub-ADC errors!**
- Only error term in \(D_{out}\) contains quantization error of last stage
- So why do we care about sub-ADC errors? ➢ Go back to two stage example
Sub-ADC Errors

\[ D_{out} = V_{in,ADC} + \frac{\varepsilon_{in}}{\prod_{j=1}^{n} G_j} \]

Grows outside ½ LSB bounds

Sub-ADC Errors

Ideal 2-Stage Pipelined ADC

2-Stage Pipelined ADC with Coarse ADC Comp. Offset

0111 (7)

1000 (WRONG)
1st-Stage Comparator Offset

Problem: $V_{res1}$ exceeds 2nd pipeline stage overload range

First stage ADC Levels: $(\Delta = 1)$
Ideal comparator threshold: -1, 0, +1
Comparator threshold including offset: -1, 0.3, +1

Three Ways to Deal with Errors...

- All involve "sub-ADC redundancy"
- Redundancy in stage that produces errors
  - Choose gain for 2nd stage < $2^{B_1}$
  - Higher resolution sub-ADC
- Redundancy in succeeding stage(s)
(1) Gain for $2^{\text{nd}}$ stage < $2^{B_1}$

- Choose $G_1$ slightly less than $2^{B_1}$

- Effective stage resolution becomes non-integer

$B_{1_{\text{eff}}} = \log_2 G_1$

Ref: A. Karanicolas et. al., JSSC 12/1993

Correction Through Redundancy

- "enlarged" residuum still within input range of next stage

If $G_1=2$ → only 1 Bit resolution from first stage (3 Bit total) →

Overall ADC Transfer Curve
(2) Higher Resolution Sub-ADC

- Keep $G_1$, precise power of two (e.g. keep $G_1 = 4$)
- Add extra decision levels in sub-ADC (e.g. add 1 extra bit to 1st stage)
- E.g. $B_1 = B_{\text{ref}} + 1$

Ref: Singer et. al., VSLI1996

(3) Over-Range Accommodation Through Increase in Following Stage Resolution

- No redundancy in stage with errors
- Add extra decision levels in succeeding stage

Ref: Opris et. al., JSSC 12/1998
Redundancy

- The preceding analysis applies to any stage in an n-stage pipeline.
- Can always look at pipeline as a single stage + backend ADC.

In literature, sub-ADC redundancy schemes are often called "digital correction" – a misnomer!
- No error correction takes place.
- We can tolerate sub-ADC errors as long as:
  - The residues stay "within the box", or
  - Another stage downstream "returns the residue to within the box" before it reaches last quantizer.

Let's calculate tolerable errors for popular "1.5 bits/stage" topology.
1.5 Bits/Stage Example

- Comparators placed strategically to minimize overhead
- $G=2$
- $B_{\text{eff}}=\log_2 2 = 1$
- $B=\log_2 (2+1) = 1.589...$

Ref: Lewis et. al., JSSC 3/1992

3-Stage 1.5-bps Pipelined ADC

- All three stages
- Comparator with offset
- Overall transfer curve
  - No missing codes
  - Some DNL error

Amplifier Offset

- Input referred converter offset – usually no problem
- Equivalent sub-ADC offset - easily accommodated through redundancy

Gain Errors

\[ D_{out} = V_{in,ADC} + \varepsilon_{q1} \left( 1 - \frac{G_1 + \delta}{G_{d1} + \delta} \right) \]
\[ + \varepsilon_{q1} G_2 \left( 1 - \frac{1}{G_{d2}} \right) + \ldots + \varepsilon_{q(n-1)} G_{n-1} \left( 1 - \frac{1}{\prod_{j=1}^{n-1} G_{d(j-1)}} \right) + \varepsilon_{qn} \frac{1}{\prod_{j=1}^{n} G_{dj}} \]
Gain Errors

• Gain error can be compensated in digital domain – "Digital Calibration"

• Problem: Need to measure/calibrate digital correction coefficient

• Example: Calibrate 1-bit first stage

• Objective: Measure G in digital domain
### ADC Model

$V_{in}$

1-bit ADC

1-bit DAC

$V_{ref}$

$V_{res1} = G \cdot (V_{in} - V_{DAC})$

$V_{DAC} (D = 0) = 0$

$V_{DAC} (D = 1) = V_{ref} / 2$

### Calibration – Step 1

$V_{in} = \text{const.}$

1-bit ADC

1-bit DAC

$MUX$

$\ast 1 \ast$

$V_{ref}$

$V_{res1}^{(1)} = G \cdot (V_{in} - V_{ref} / 2)$

$D_{back}^{(1)} = G \cdot \frac{(V_{in} - V_{ref} / 2)}{V_{ref}} \rightarrow \text{store}$
Calibration – Step 2

\[ V_{\text{res1}}^{(2)} = G \cdot (V_{\text{in}} - 0) \]
\[ D_{\text{back}}^{(2)} = G \cdot \frac{(V_{\text{in}} - 0)}{V_{\text{ref}}} \rightarrow \text{store} \]

Calibration – Evaluate

\[ D_{\text{back}}^{(1)} = G \cdot \frac{(V_{\text{in}} - V_{\text{ref}} / 2)}{V_{\text{ref}}} \]
\[ -D_{\text{back}}^{(2)} = G \cdot \frac{(V_{\text{in}} - 0)}{V_{\text{ref}}} \]
\[ D_{\text{back}}^{(1)} - D_{\text{back}}^{(2)} = \frac{1}{2} \cdot G \]
Accuracy Bootstrapping

\[ D_{\text{out}} = V_{\text{in,ADC}} + \varepsilon_q(1 - \frac{G_1}{G_d}) + \frac{\varepsilon_{q2}}{G_d} \left( 1 - \frac{G_2}{G_d} \right) + \ldots + \left( \frac{\varepsilon_{q(n-1)}}{G_d} \right) \left( 1 - \frac{G_{(n-1)}}{G_d} \right) + \frac{\varepsilon_{qn}}{G_d} \]

- Highest sensitivity to gain errors in front-end stages

"Accuracy Bootstrapping"

\[ V_{\text{in}} \]

Stage 1 \quad \text{Stage 2} \quad \text{Stage 3} \quad \text{Stage k} \quad B_n \text{ bits}

\[ \rightarrow \text{Calibration in opposite direction...} \]

Ref:
L. Singer et al., "A 12 b 65 MSample/s CMOS ADC with 82 dB SFDR at 120 MHz," ISSCC 2000, Digest of Techn. Papers., pp. 38-9
DAC Errors

- Can be corrected digitally as well
- Same calibration concept as gain errors

DAC Calibration – Step 1

- $\epsilon_{DAC}(0)$ equivalent to offset - ignore
DAC Calibration – Step 2…2^{B_1}

- Stepping through DAC codes 1…2^{B_1}-1 yields all incremental correction values

Calibration Hardware

- Digital is "free" and easier to build than precise analog circuits...

Amplifier Nonlinearity

\[ V_{\text{IN1}} \xrightarrow{\text{ADC}} V_{\text{RES1}} \xrightarrow{\text{DAC}} V_{\text{OS}} \xrightarrow{a_3 V^3} V_{\text{RES1}} \]

\[ \Delta \]

Measurement Results

(a) without calibration

(b) with calibration

INL [LSB] vs. Code

RNG=0
RNG=1